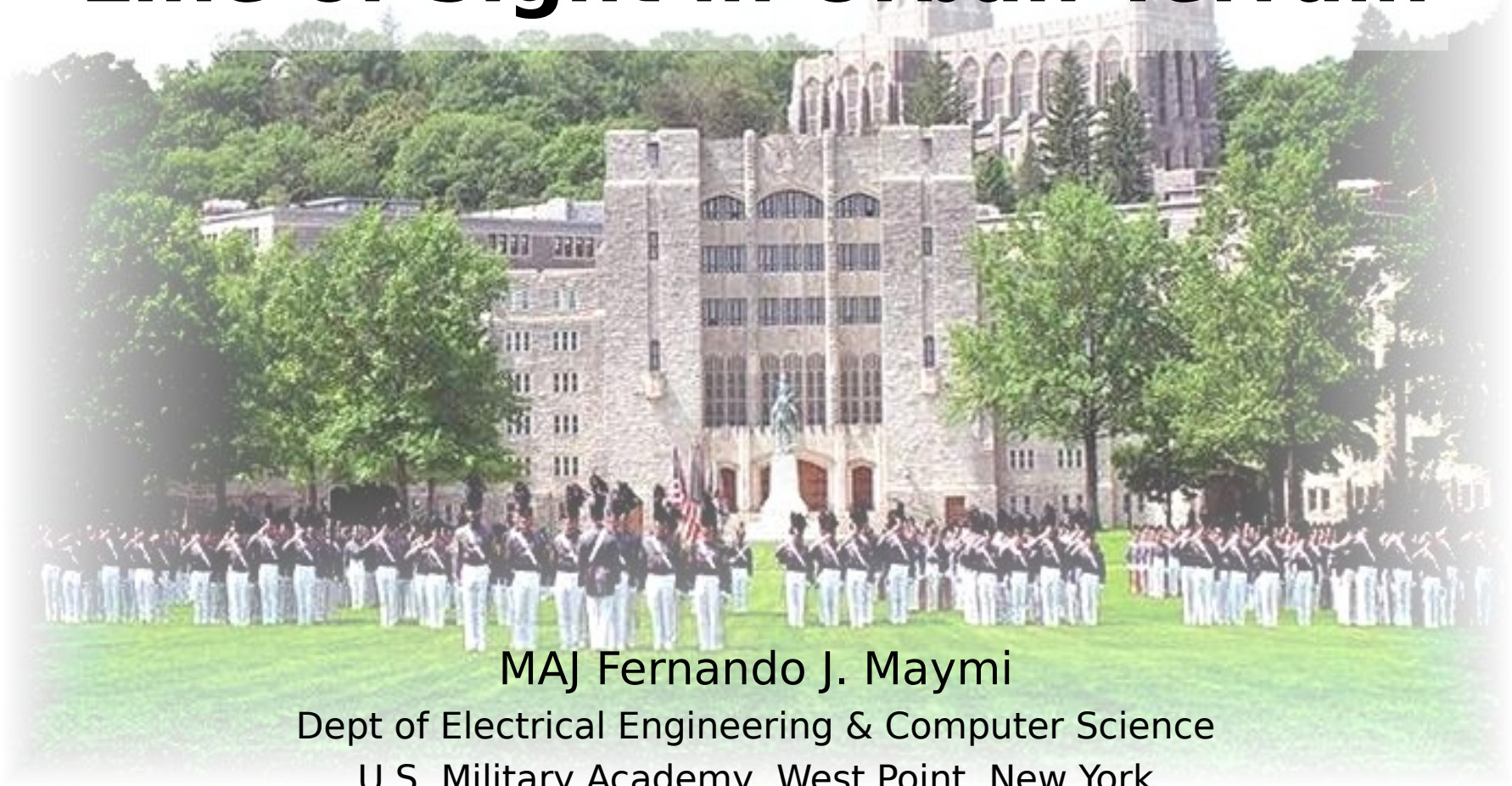
Calculating Probability of Line of Sight in Urban Terrain



MAJ Fernando J. Maymi
Dept of Electrical Engineering & Computer Science
U.S. Military Academy, West Point, New York
fernando.maymi@usma.edu

Duty, Honor, Country

Agenda

- Academic Background
- Problem Description
- Team Make-up
- Analysis
- Design
- Implementation
- Verification & Validation
- Lessons Learned

Academic Background

- Computer Information Systems Design I and II
  - Capstone course for CS & ISE majors
  - Project-driven course
  - Students complete 2 iterations of SDLC

Academic Background

- All majors in D/EE&CS must complete a year-long design project
- Real-world clients
- Advanced Individual Academic Development
  - David Bergeron at TRAC Monterey

Problem Description

- Line of sight in urban terrain
  - Computationally expensive
  - Required for simulations of urban combat operations
- Outcomes from combat models are highly correlated with sensor performance representation

The Team

Alan Bairley Computer Science

David Bergeron ISE (CS focus)

David Fierner Geospatial Eng.

Manuel Ramos ISE (SE focus)

Team's Accomplishments

- Developed a tool that uses a simple representation of urban terrain to estimate probabilities of line of sight
- Verified results from the tool using urban terrain data and sample sensor locations
- Begun validation by comparing the tool results to field data collected at West Point

Requirements

- Calculate PLOS given terrain data and sensor locations
- Produce exportable outputs
- Quantify the difference between calculated and empirical results
- Platform-independence

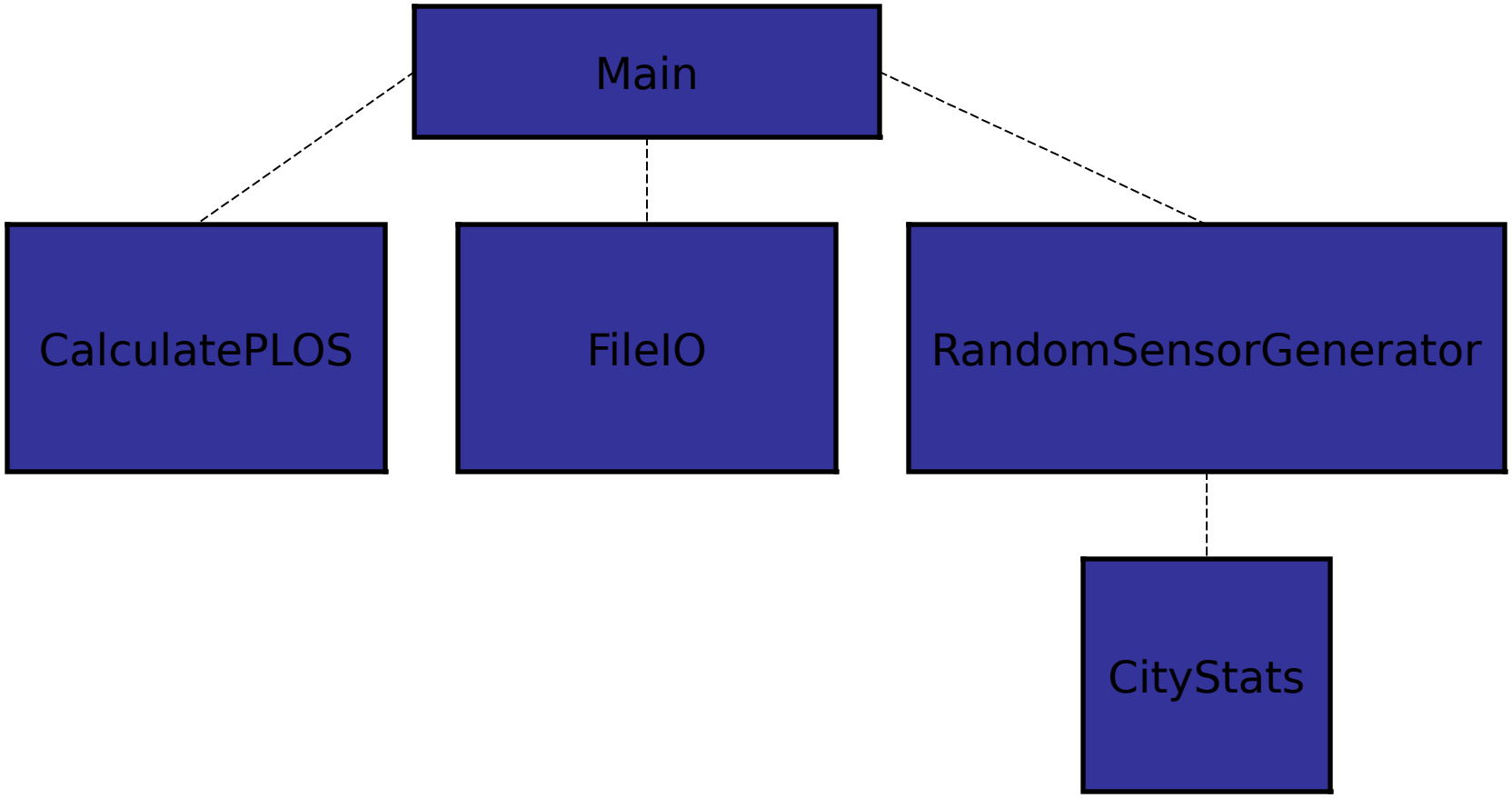
Assumptions

- Expert user
- No terrain exists outside given data
- Curvature of the Earth is negligible
- Synthetic terrain accurately depicts Urban Terrain Zone (UTZ)

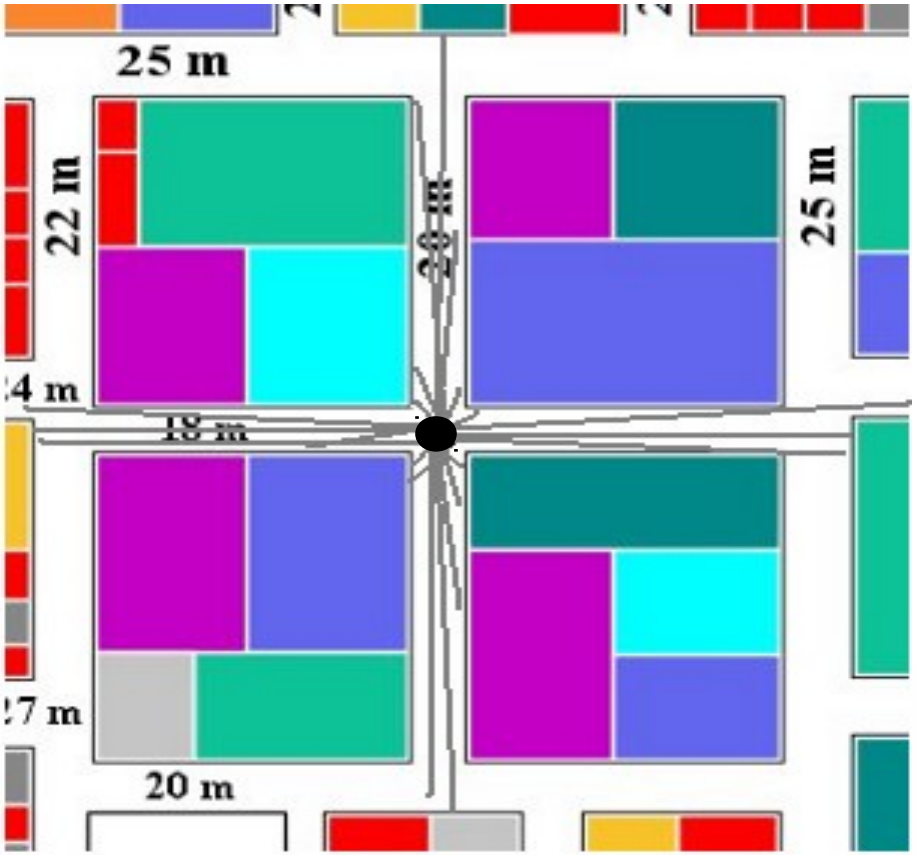
Design Considerations

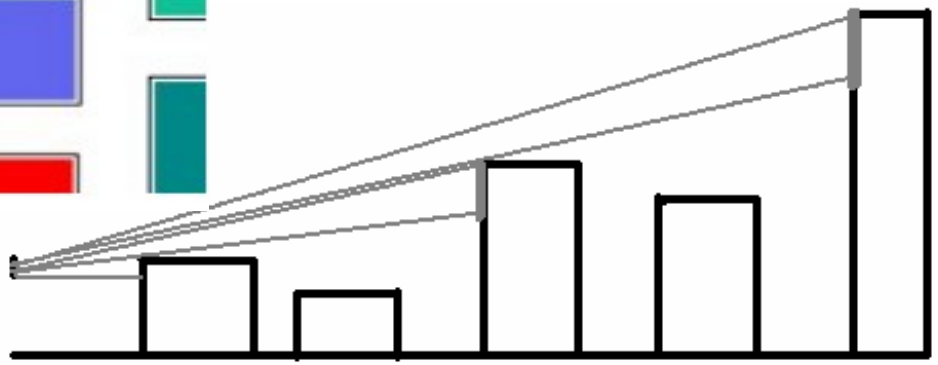
- User Interface (command prompt)
- Modular
- Data Storage
- User Inputs
- Testing Levels

Design


Main
CalculatePLOS
FileIO
RandomSensorGenerator
CityStats

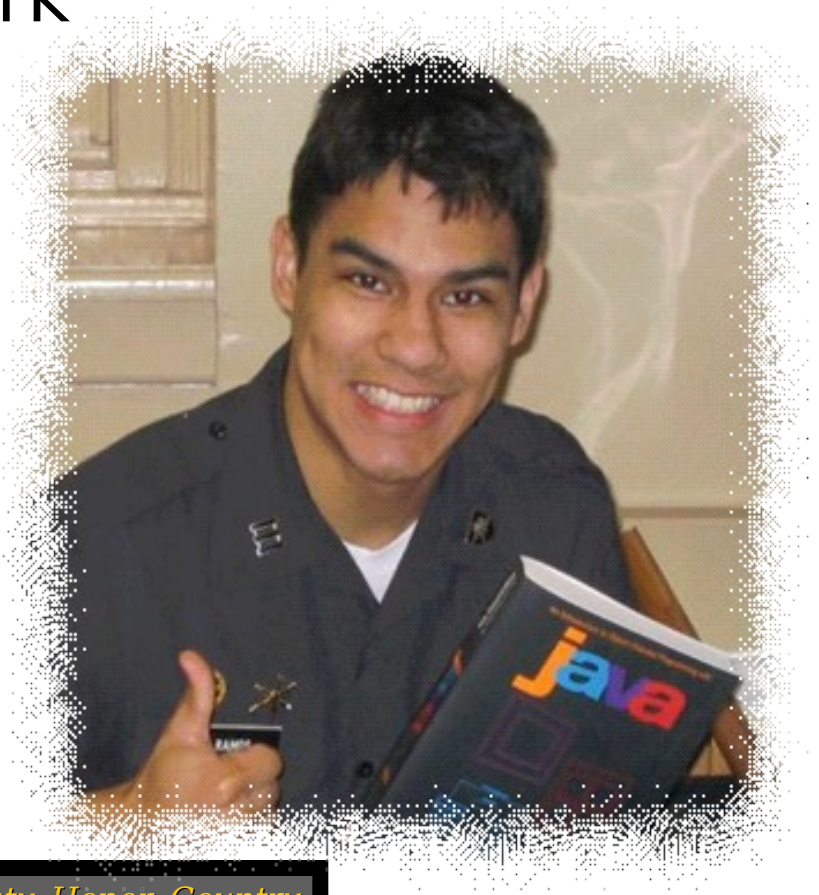
Methodology





Implementation

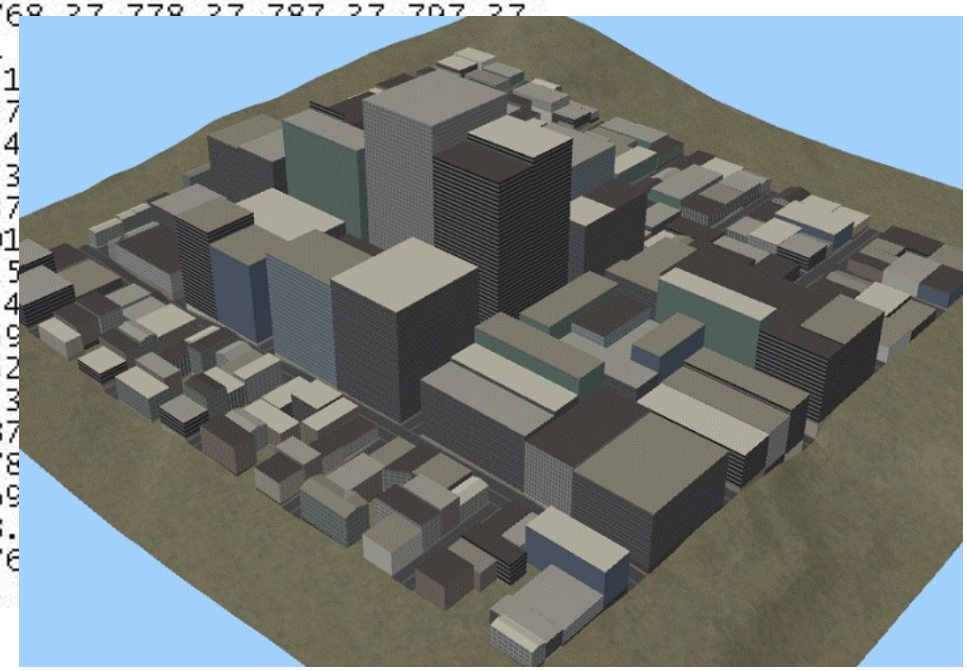
- Java
- Concurrent work
- Heap space
- -Xprof



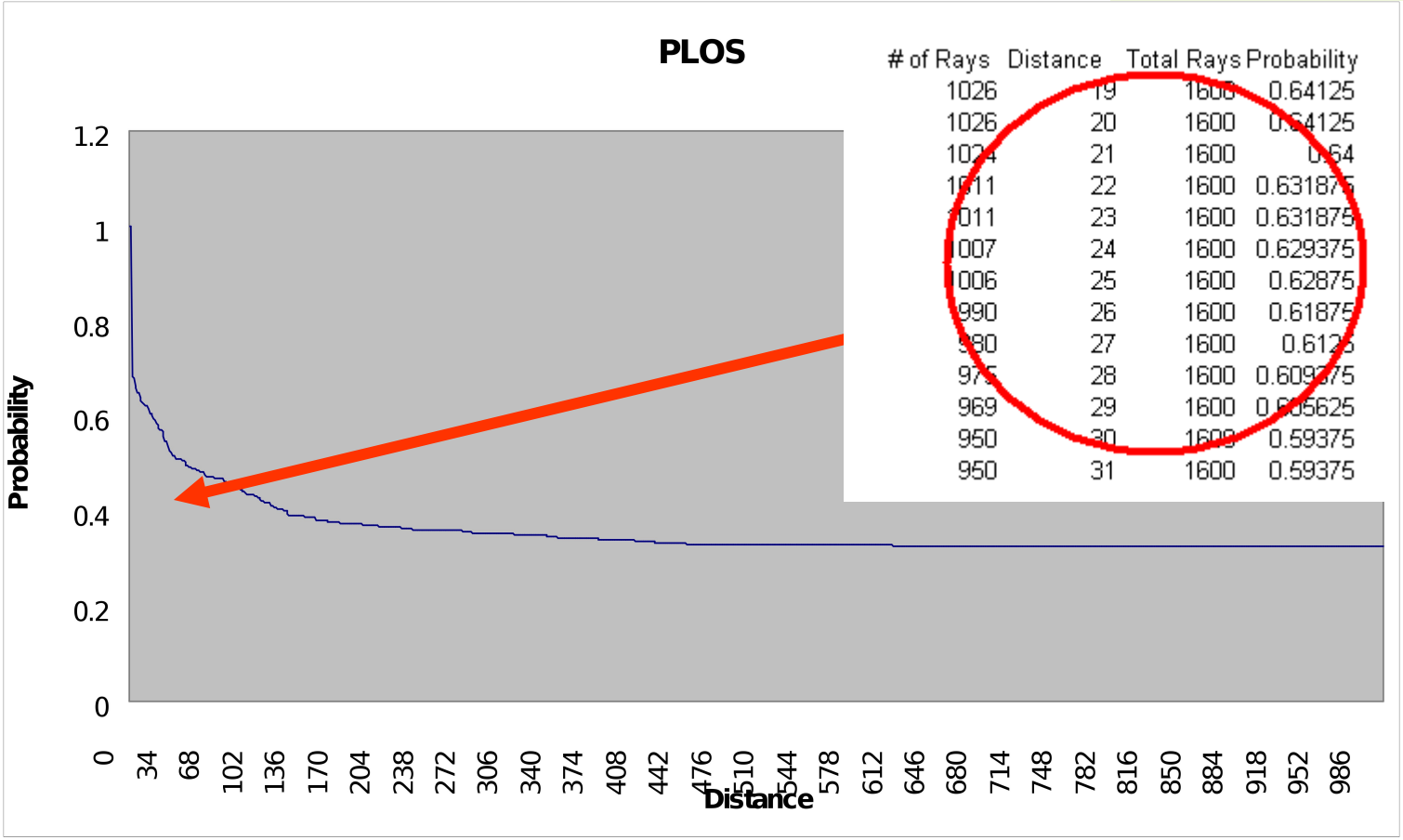
Data Considerations

- Configuration
- Terrain
- Sensors
- Output

```
106.145 106.16 106.175 106.19 106.205
4 84.672 84.349 84.027 83.705 83.383 8
51.152 50.934 50.715 50.496 50.278 50.
63 41.215 41.167 41.119 41.071 41.023
8.846 38.812 38.778 38.717 38.617 38.5
37.758 37.768 37.778 37.787 37.797 37
.234 95.161
69 105.201 1
096 77.861 7
.77 47.659 4
084 40.036 3
05 37.934 37
38.007 38.01
4 94.51 94.5
04.622 104.4
72.788 72.59
46.579 46.52
349 39.351 3
37.872 37.87
37.882 37.78
96.585 96.69
8 99.101 98.
66.062 65.76
.585 44.504
```



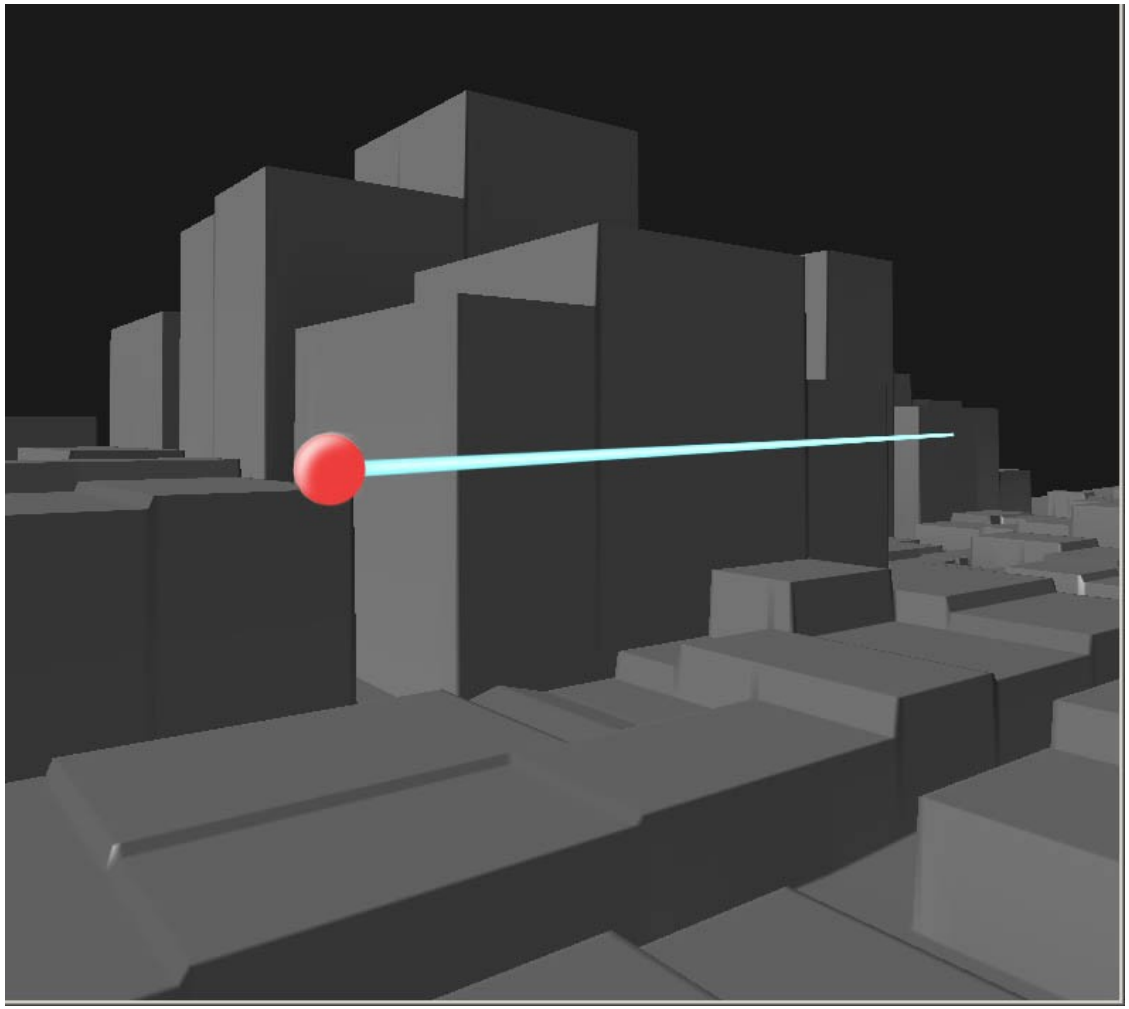
Results



# of Rays	Distance	Total Rays	Probability
1026	19	1600	0.64125
1026	20	1600	0.64125
1024	21	1600	0.64
1011	22	1600	0.631875
1011	23	1600	0.631875
1007	24	1600	0.629375
1006	25	1600	0.62875
990	26	1600	0.61875
980	27	1600	0.6125
975	28	1600	0.609375
969	29	1600	0.605625
950	30	1600	0.59375
950	31	1600	0.59375

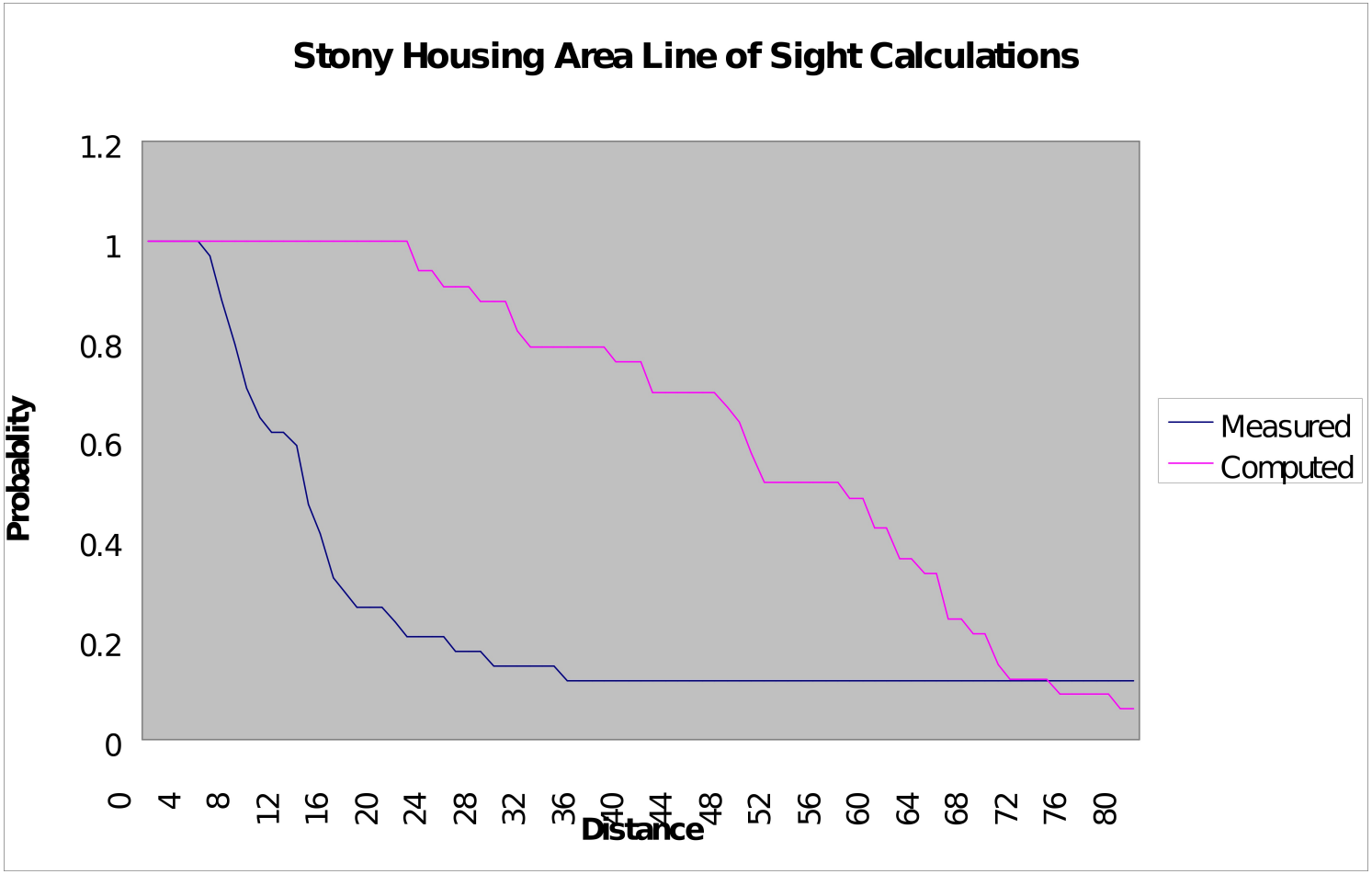
Aggregate based on 10 randomly placed ground sensor scanning from 0 to 10 degrees above the horizontal axis in City Core Data

Visual Verification



Field Validation

Validation Results

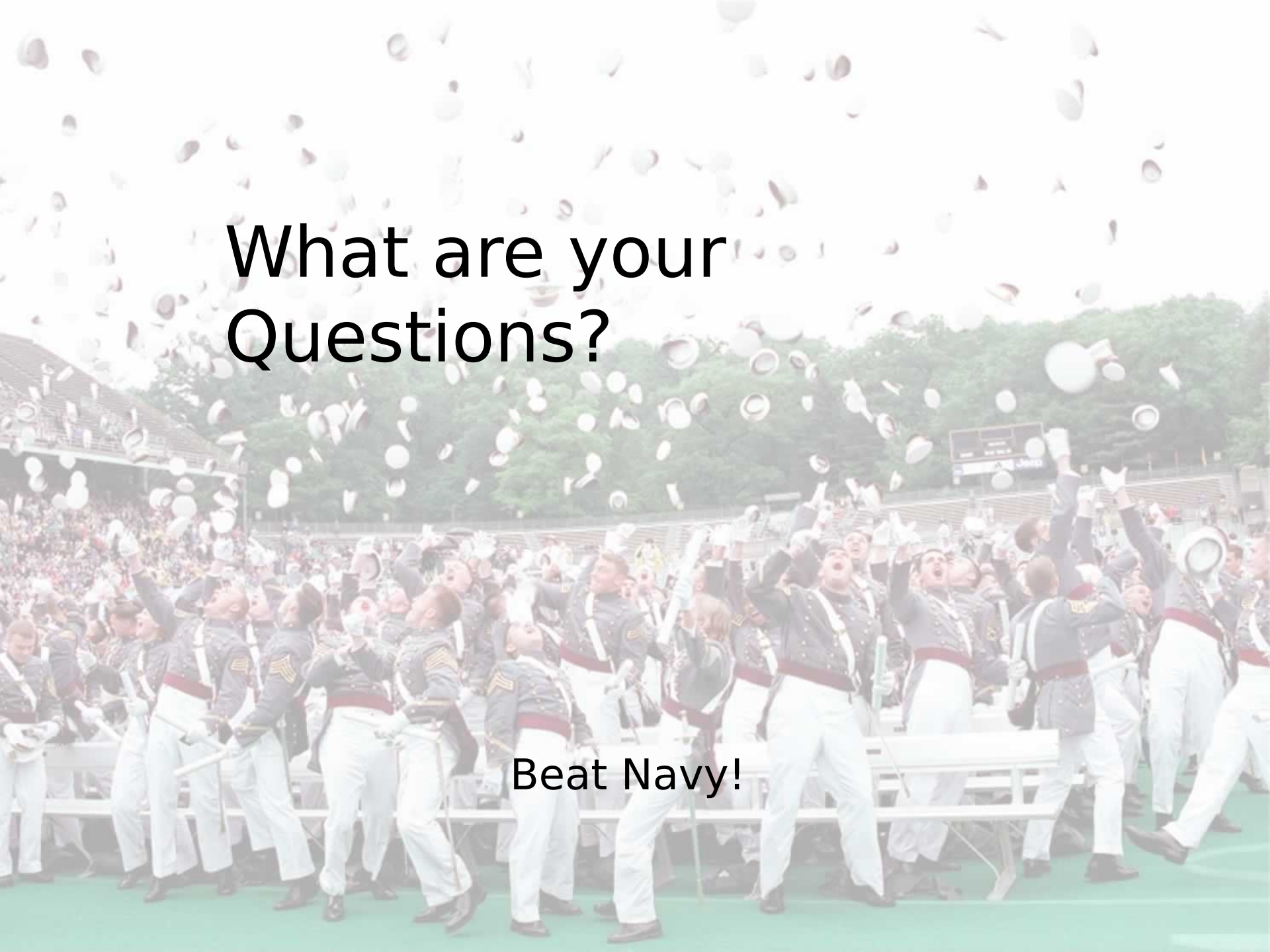


Validation Issues

- Reflective surfaces required to measure ranges
- Intervening Features
  - Foliage
  - Parked cars
  - Fences

Future Work

- Integrate UTZ Viewer and PLOS Calculation
- Refine algorithms to further improve performance
- Support multiple PLOS algorithms



What are your Questions?

Beat Navy!